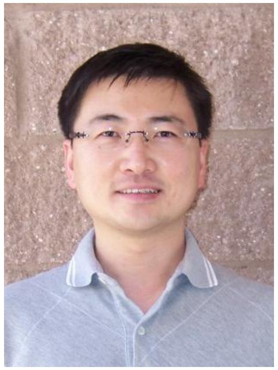


Tim Kim, Ph.D.

Principal Research Engineer

Tim Kim, Ph.D. is a Principal Research Engineer at Ridgetop Group. He has more than 10 years of experience in autonomic computing, anomaly detection and root cause analysis systems, troubleshooting, electronic prognostics, and data analysis.

Troubleshooting Analysis and Decision Support in Complex Applications

Tim (Byoung Uk) Kim, Ph.D.

July 25th, 2012

Outline

- Introduction to Troubleshooting Analysis
- Troubleshooting Requirements
- Causal Bayesian Networks
- Testbed
- Training Results
- Case Study
- Conclusion
- Q&A

Introduction to Troubleshooting Analysis

Complex systems, such as aircraft, require rigorous routine inspection and maintenance to ensure the health of the plane’s numerous mechanical and electronic systems. While vital, this constant process has seen significant cost increases such as:

- Labor
- Parts
- Aircraft downtime rise
- Increases in operation costs

These cost increases provide the motivation to develop an effective troubleshooting and decision support system for complex system faults/failures.

Introduction to Troubleshooting Analysis (cont.)

Due to the complexity of system failure diagnosis and troubleshooting, existing tools focus on individual components of the system and neglect the interactions between other components of the system.

- Compared to standard troubleshooting flow diagrams, Ridgetop's causal approach has strong capabilities by dealing with:
  - Multiple states on nodes
  - Representing dependencies among failures more explicitly
  - Providing more complex relationships between causes and effects

Troubleshooting Flow Diagram (TFD)

- The TFD is a commonly used method for quantitative risk modeling
- It is a popular methodology for evaluating failure occurrences in safety-critical systems in a top-down fashion
- The system failure is often represented by the event at the top; it is decomposed into basic events that describe detailed causes and basic components' failures
- Logic gates like AND or OR provide the logic expressions among different failure events
- Given the probabilities of basic events and the logic structure of the tree, the probability of system failure can then be calculated

General Features of Bayesian Networks (BN)

- BN are directed acyclic graphs (DAG) for representing the joint distribution and reasoning under uncertainty
- Reasoning under uncertainty is the capability of representing and extrapolating data with uncertainty, due to noise or errors
- Each node is assigned a random variable with certain probability distribution in describing a particular event
- Probability distribution can fully capture the uncertain information of the data
- Conditional probability tables (CPT) are specified for each pair of connected nodes in quantifying the dependency strength

Benefits of an Approach

- CPT in Bayesian networks can cover more complex relationships between causes and effects
  - In the TFD, simple logic operations like AND, OR are used
  - In reality, there might be other unknown causes which the model fails to cover that will also determine the health status of the overall system

Benefits of an Approach (cont.)

- Nodes in Bayesian networks can deal with multiple states
  - In the TFD, each event node only has binary values like True or False
  - For the real application (e.g., EMA), the virtual sensor can give multiple states of the component health status, such as 80%. This indicates that the component is not perfectly good but at least acceptable
  - In Bayesian networks, since a random variable is defined at each node, multiple discrete values can be assigned directly

Benefits of an Approach (cont.)

- Bayesian networks can represent dependencies among failures more explicitly
  - TFD assumes that failure events are independent. In practice, however, some component failures can be dependent
  - In the TFD, it is hard to model this dependency situation, while in Bayesian networks, simply using arcs can express this cause-and-effect information flow

Requirement of Troubleshooting Reasoning

Three requirements:

- Anomaly Detection
- Failure Diagnosis and Isolation
- Failure Prognosis

Anomaly Detection

- This sub-module detects the failure of components, subsystems or even systems, and indicates that "something failed" in the system
- Statistical process control concepts and charting techniques are utilized in this sub-module to monitor the system performance based on the performance measurement data collection from the component, subsystem, and system level

Failure Diagnosis and Isolation

- This sub-module answers the questions of “what failed” and “why it failed.”
- It locates the component(s) and subsystem(s) that caused the system failure and identifies the root cause of the failure by mapping the statistical patterns extracted from multivariate data with the engineering knowledge representation of failure physics

Failure Prognosis

- This sub-module answers the question “What will fail?”
- Based on the normal performance measures identified in anomaly detection and the mathematical cause-effect model learned in failure diagnosis, the prediction of potential failure, given current data and current status of the component/subsystem, can be implemented

Causal Bayesian Network

Two Approaches

Constrained-based

- Uses the conditional independency information in determining the Bayesian network structure
- More suitable for incorporating domain constraints

Approximation-based

- Convert the learning problem into an optimization problem by adding the scoring function to the structure

Causal Bayesian Network Structure

- The output of the CBN training module is defined as an adjacency matrix:

$$M_{n\times n}=\begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix}$$

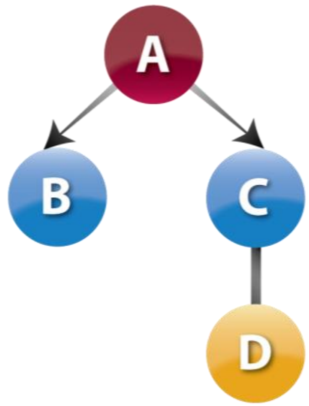
where n refers to the number of nodes

- Edge information as follows:

$$m_{ij}=\begin{cases} 0 & \text{No connection between node i and j} \\ 1 & \text{Undirected connection between node i and j} \\ -1 & \text{Directed connection from node i to j} \end{cases}$$

Causal Bayesian Network Structure

- Example of the adjacency matrix: node A is the common cause of nodes B and C while C and D have only an undirected connection



$$M_{4\times 4} = \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- If viewing the adjacency matrix from a row perspective, each row represents its effect variables information
- Likewise, a column perspective gives cause information for that particular node
- Taking the 1st row as an example, the 2nd and 3rd columns have -1, which means B and C are two effects for node A
- Taking the 3rd column as an example, only the 1st row has -1, which means C has only one cause
- The element value of 1 only gives connection information but no cause-and-effect relationship

Skeleton Identification Flow

1. Begin with a complete flow diagram D'
2. $i = 0$
3. Repeat
4.   For each $A \in \mathrm{A}$
5.   For each $B \in ADJ_A$
6.   Test whether $\exists S \subseteq ADJ_A - \{Y\}$ with $|S| = i$
                and $I(A, B \mid S)$
7.   If this set exists
8.   Make $S_{AB} = S$
9.   Remove $A - B$ link from D'
10. $i = i + 1$
11. Until $|ADJ_A| < i, \forall A$

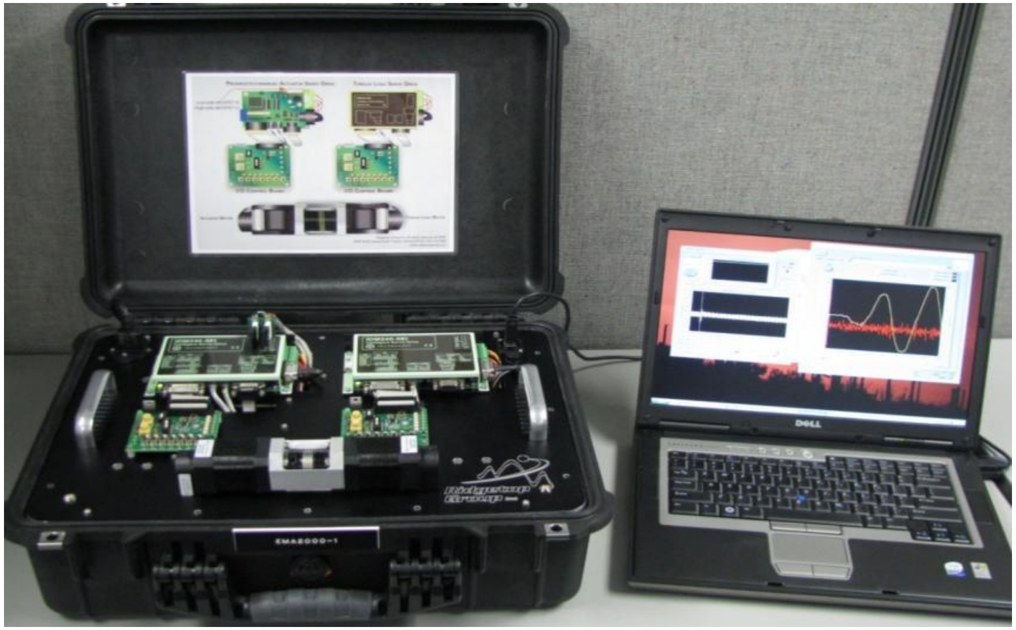
Where A is defined as a set of variables. Each Ai represents a node in the CBN. $I(A, B \mid S)$ (independence test) is computed between two nodes to determine whether the link between them should be removed or not.

Edge Orientation Flow

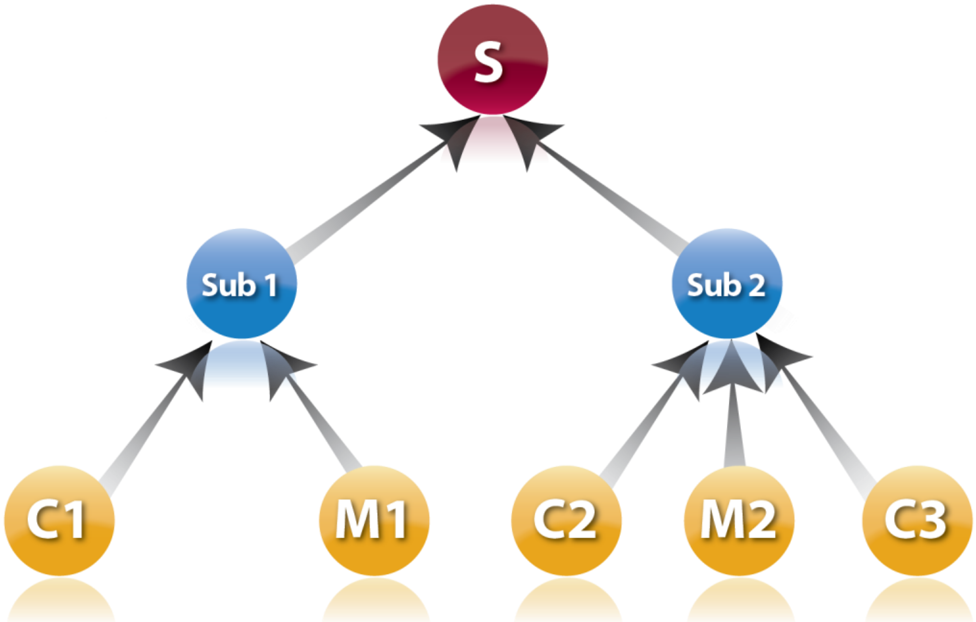
- After skeleton identification, the original graph with all nodes connected is transformed into a graph with only conditional dependent nodes connected
- Edge orientation attempts to determine the direction of these undirected arcs
- Basic procedures can be described as follows:
  - Creating V-structure: Looking for sets of three variables {X, Y, Z} such that X-Z and Y-Z. If $Z \notin S_{XY}$, orient $X \rightarrow Z$ and $Y \rightarrow Z$
  - Orient the remaining undirected edges with two basic principles: no cycles and no new V-structure
  - Arbitrarily orient the remaining undirected edges

Testbed - EMA System

- A fault-enabled 24 VDC supply to power the three phases of the electromechanical actuator servo drives and integrate the switch-mode power supply (SMPS) with a high-speed DAQ unit
- Leverage data from a complex mechanical/electrical EMA system-of-systems including various components, utilizing direct experience and knowledge



Structure of Three-level System



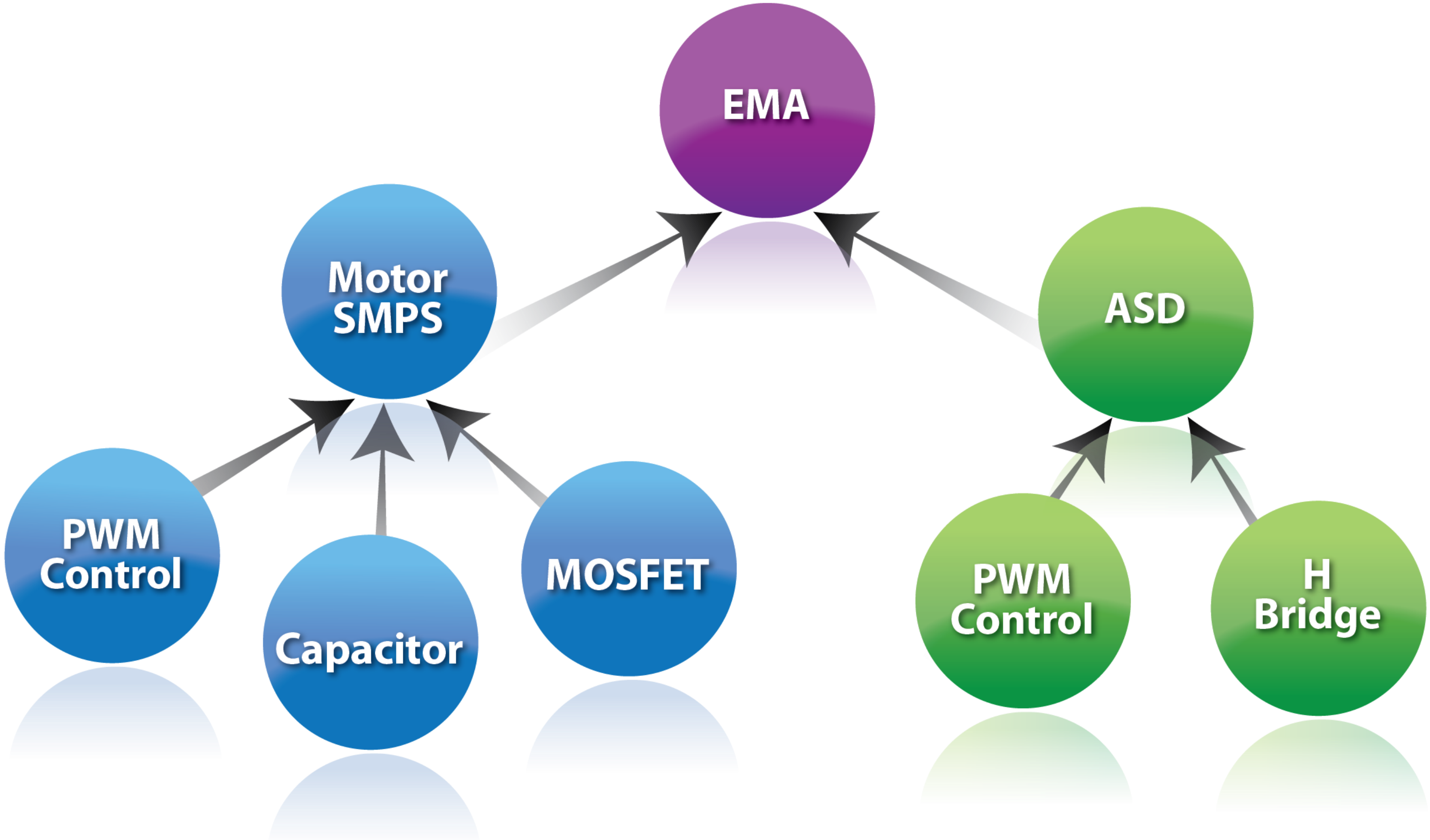
- Prior probabilities and conditional probability tables are acquired from data and domain knowledge
- Learned CBN will be compared with the true structure assumed at the beginning. If the learned Bayesian network is the same as the predefined structure, the capabilities of Bayesian network training from data could be validated

Training Results

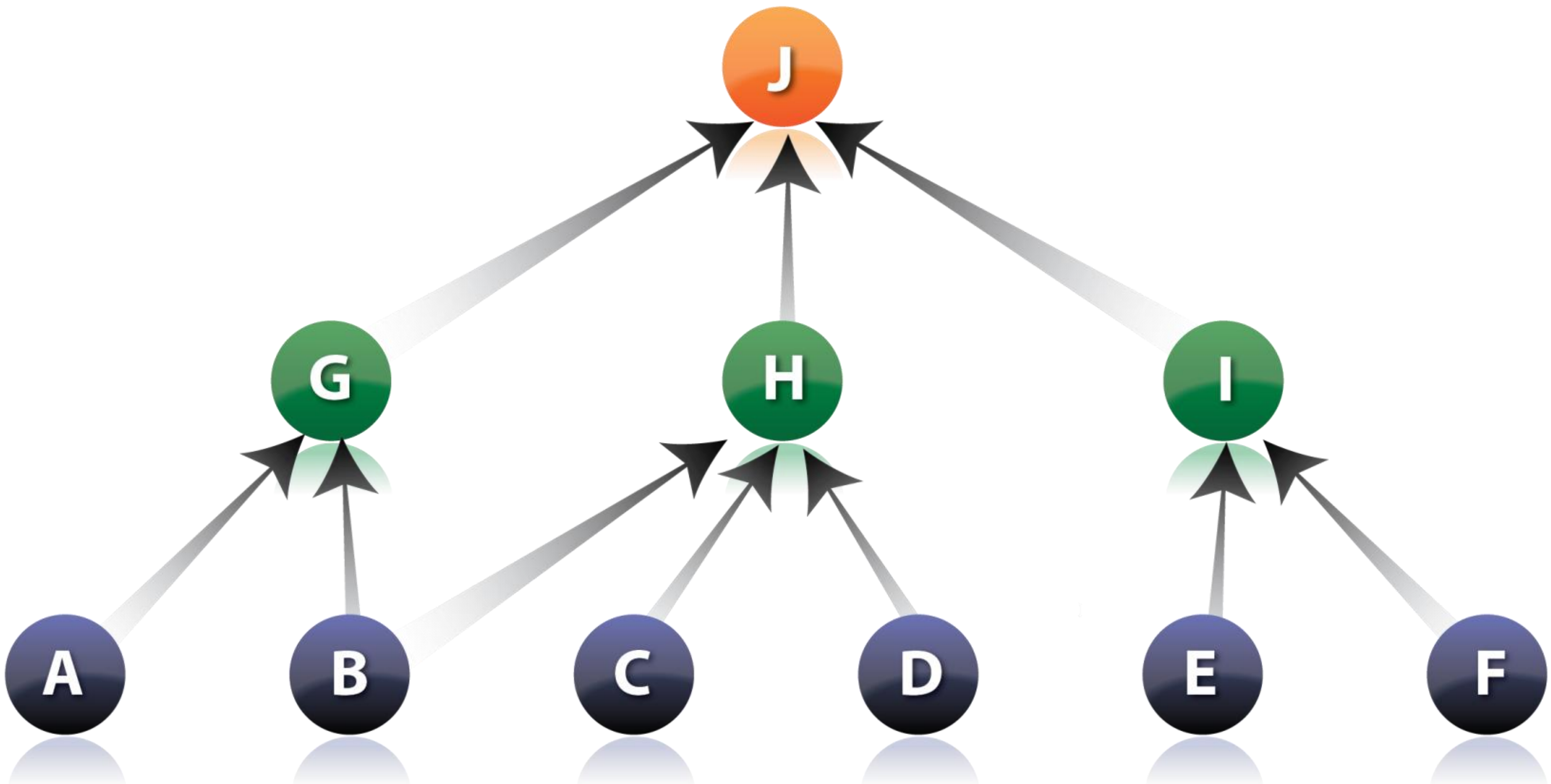
- Training results of the simulated 2000 observations
- A “0” refers to no connection while a “-1” indicates that there exists an oriented connection
- Take the -1 at row 1 and column 6 as an example. This indicates there is an oriented connection from node 1 to node 6; that is, there is a directed arc from node “PWM Control in the Motor SMPS” (node 1) to node “Motor SMPS subsystem” (node 6)

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	-1	0	0
2	0	0	0	0	0	-1	0	0
3	0	0	0	0	0	-1	0	0
4	0	0	0	0	0	0	-1	0
5	0	0	0	0	0	0	-1	0
6	0	0	0	0	0	0	0	-1
7	0	0	0	0	0	0	0	-1
8	0	0	0	0	0	0	0	0
9								

Implementation Results of the Constructed CBN

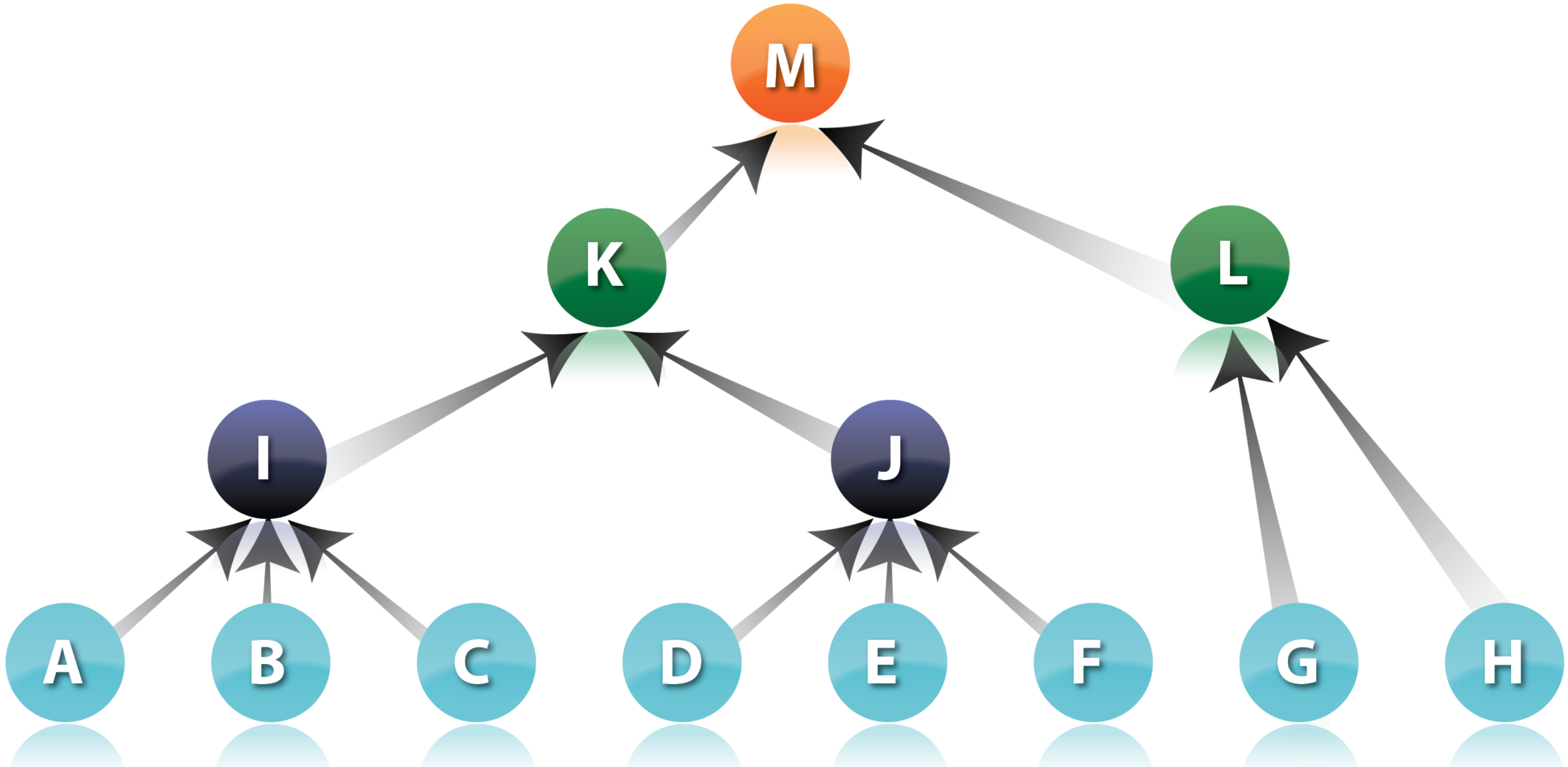


Results of Case 1, with Three Levels of Faults



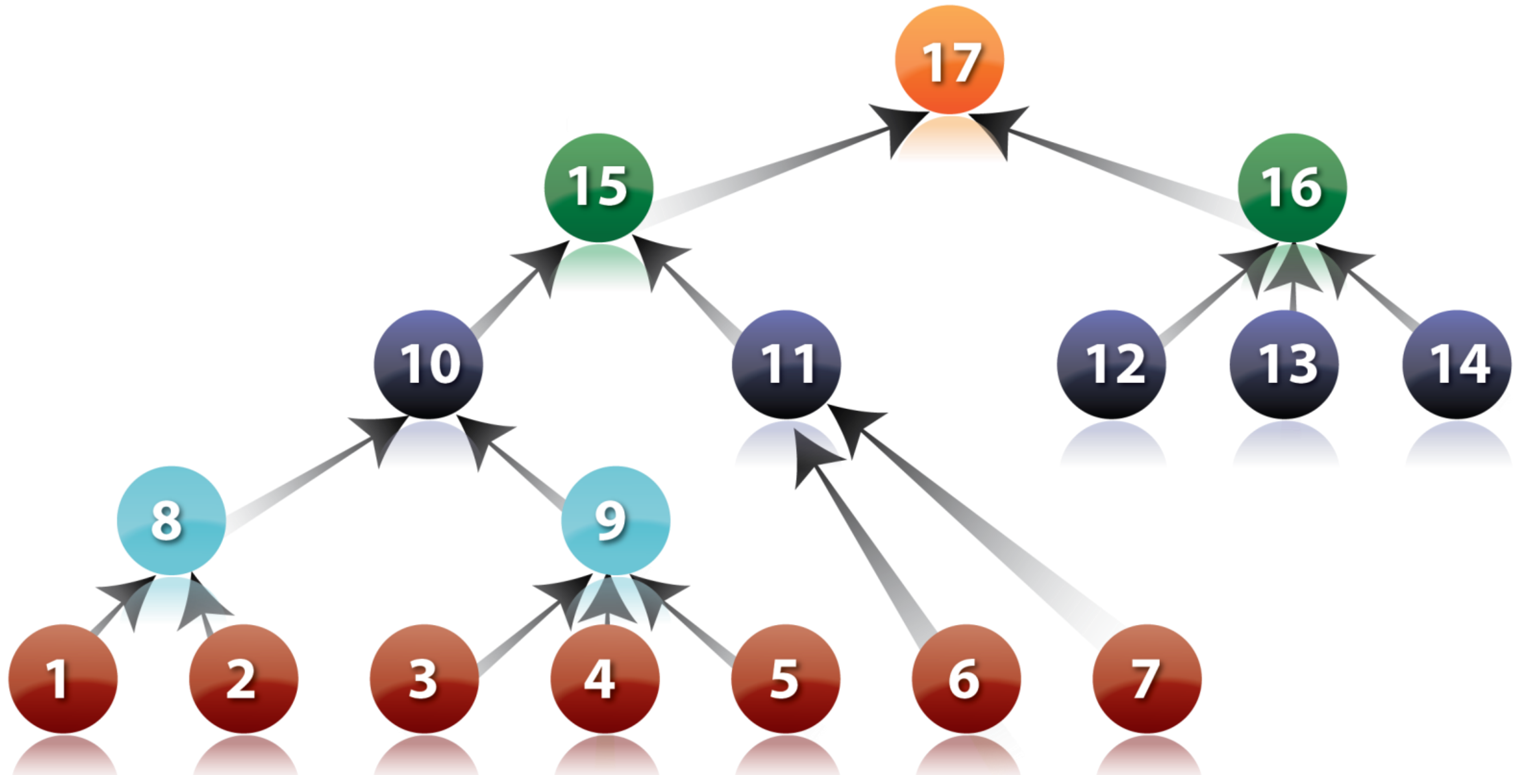
- Node “B” is the common cause for more than one subsystem
- For each level, there is no limit for the number of nodes

Results of Case 2, with Four Levels of Faults



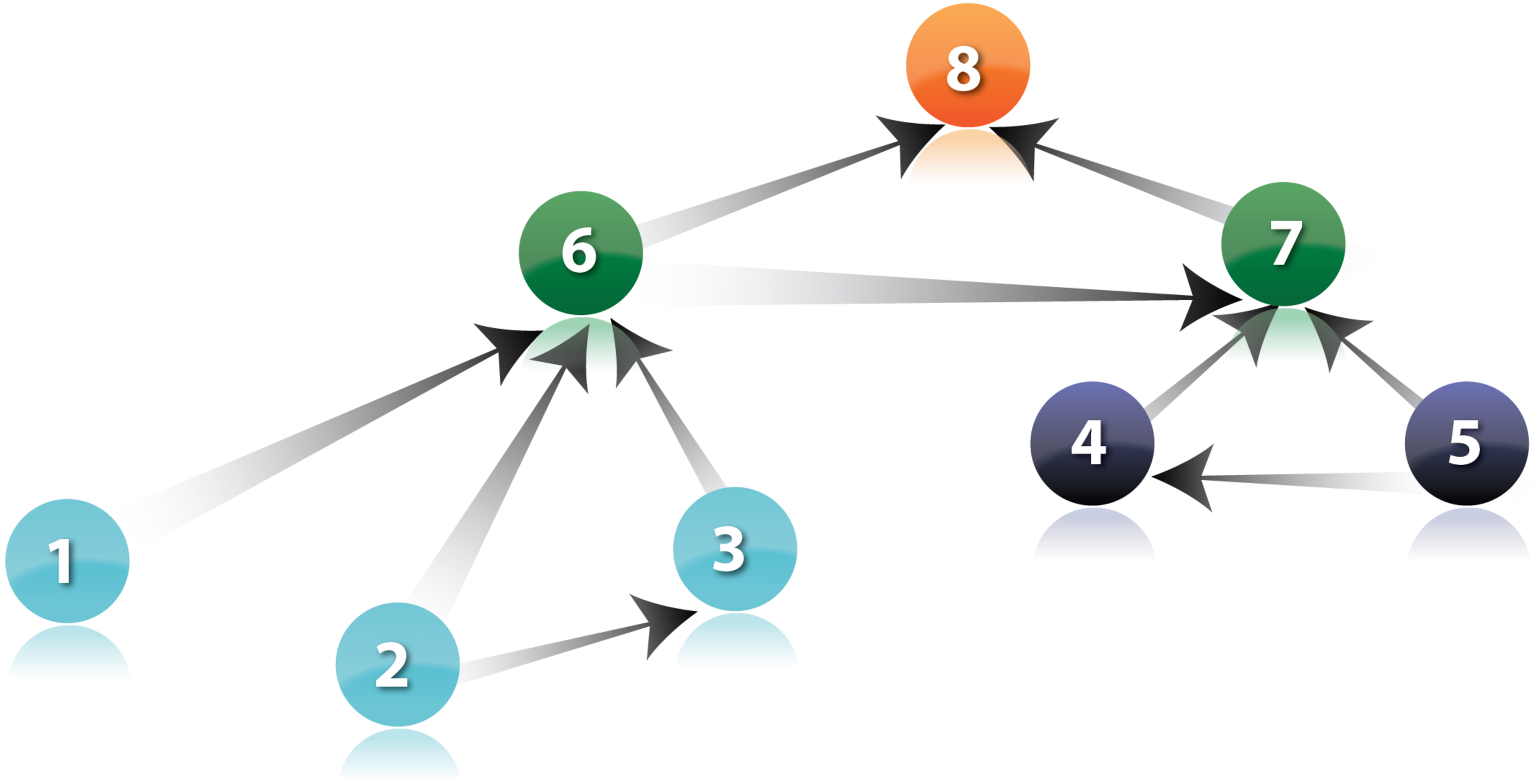
- Nodes “G” and “H” as an example, their components can directly influence the third level beyond the second level
- If viewing this CBN from a graphical tree perspective, the depth of the sub-tree can be smaller than the total depth of the overall tree

Results of Case 3, with Five Levels of Faults



- If viewing this CBN from the graphical tree perspective, the depth for each sub-tree varies from the total depth of the overall tree
- For example, node 6 and 7 directly affect the third level beyond the second level
- Node 16 has only effects given by the third level rather than all levels below the fourth level

Results of Case 4, Three Levels with Multi-level Interactions

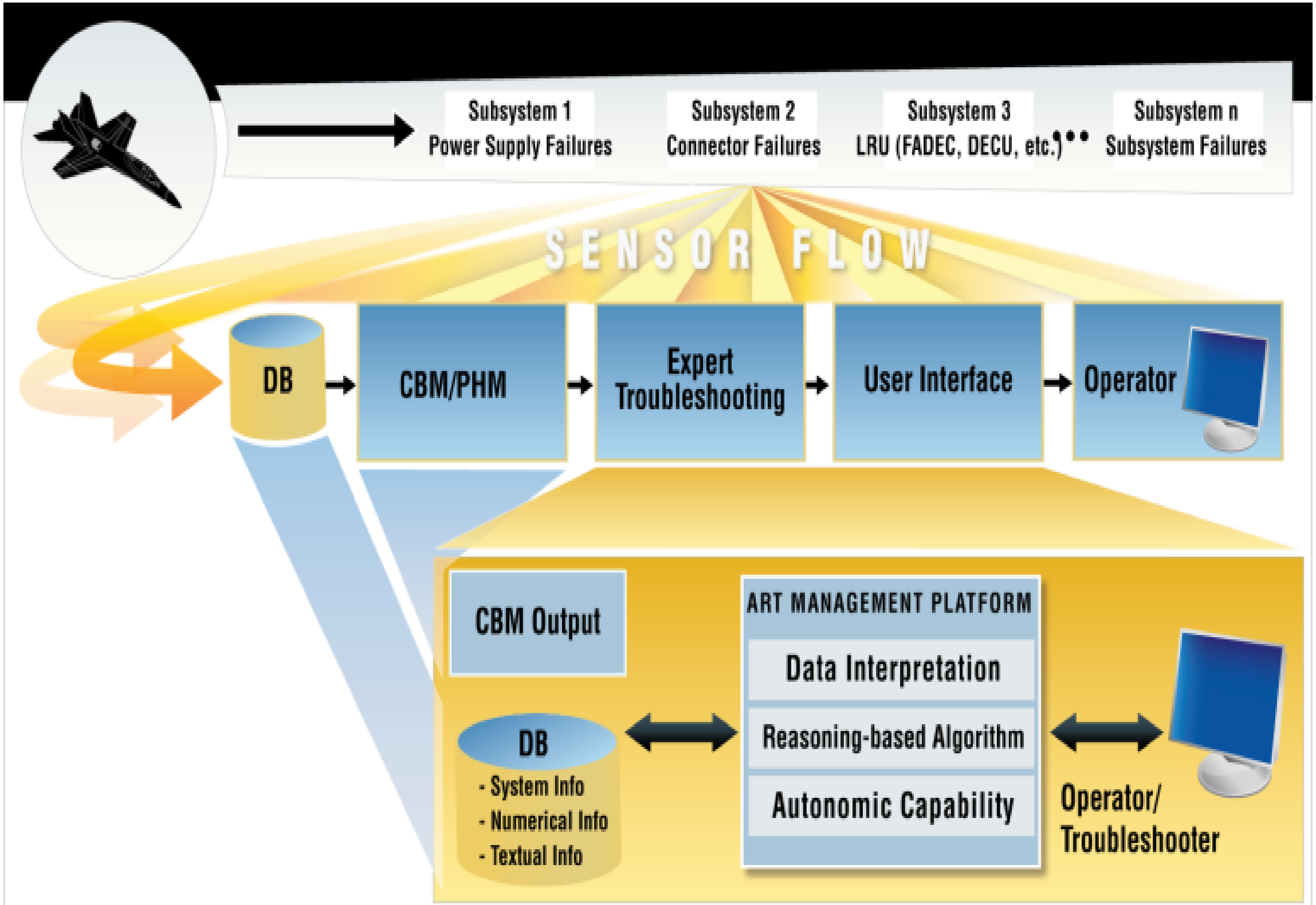


- Handle a system with interactions and failure correlations among elements at the same level
- Interactions among nodes at the same level increase the difficulties in learning the correct structure

Example of Integrated Systems Health Management (ISHM)

- As an illustration of how this can be done, consider integration with a ground-based application that provides an easy-to-understand fusion of parameters and required maintenance, so maintenance personnel can have an extra angle of insight into the systems for which they are responsible

An Example of ISHM



Conclusion

- Presented the efficacy of our causal analysis using domain knowledge and test data acquired from EMA system-of-systems integrating the SMPS with a high-speed DAQ unit
- The results from CBN implementation are validated with the true structure assumed at the beginning
- Demonstrated the flexibility and extensibility of our solutions with graphical visualization
- In our approach, causal analysis allows us to represent the component interactions and the cascaded failure/degradation propagation

Questions?

Thank You